

example above, the groups can include any combination of the aforementioned domain level group, host-group level group, host “group,” and file system “group.”

A process, e.g., executing on the aforementioned SAN manager, includes a graphical user

5 interface that displays the processor groups as a hierarchical tree. Along, for example, with the identities of the processor groups, nodes of the displayed tree list attributes of the policy defined for each respective group. As above, those attributes can include, by way of non-limiting example, whether the file system of the digital data processor is being monitored, whether the file system can be extended, a threshold value for extension, storage devices onto which the file
10 system can be extended, an extension minimum size, an extension maximum size, and an alert interval defining how often event notification is to be provided.

In related aspects, the invention provides a SAN as described above in which the process displays the hierarchical tree and its associated nodes in a first panel on a display device, such as
15 the operator/administrator console. In a second panel, the process displays interface graphical objects, e.g., list controls, dialog boxes or other editable fields, for modifying one or more attributes of a file system extension policy associated with at least a selected one of the processor groups.

20 Further aspects of the invention provide a SAN as described above in which the tree display includes at least one node identifying at least one overriden attribute, i.e., one attribute that will be overridden in the second processor group.

LUN Masking on Windows™ NT and Windows™ 2000 Hosts

Further aspects of the invention provide a storage area network (SAN) as described above that uses adapter layer filters to implement logical unit number (LUN) assignments -- or, put another 5 way, LUN masking (and unmasking) -- in the host digital data processors.

According to one such aspect of the invention, the invention provides an improved SAN of the type having one or more digital data processors, e.g., hosts of the type described above, in communication with one or more storage devices, e.g., LUNs. The host (or other digital data 10 processor) is of the type with an operating system that utilizes (i) a port driver to define a software interface between a class driver and an adapter to which one or more of the storage devices are coupled, and (ii) a class driver that claims storage devices for access, e.g., by the operating system and any applications programs executing therein, by invoking the port driver to which the host is coupled, e.g., via the interconnect fabric. The improvement comprises a 15 software filter in communication with the port driver and the class driver. That filter intervenes to block claiming of one or more selected storage devices by the class driver.

In a related aspect, the invention provides a SAN as described above where the host executes the Windows NT™ operating system and the filter blocks claiming of a selected storage device by 20 returning a failure code to the class driver in response to its invocation of the port driver for purposes of claiming that storage device.

In a further related aspect, the invention provides a SAN as described above where the host executes the Windows 2000™ operating system and the filter blocks claiming of a selected storage device by blocking claim requests by the class driver.

5 A SAN manager or other functionality is provided, according to further aspects of the invention, for transmitting to the filter identifiers, e.g., LUN IDs, of storage devices for which claiming is to be any of blocked or unblocked. In a preferred such aspect, the SAN manager or other functionality loads the filter with identifiers of storage devices for which claiming is not to be blocked, and the filter blocks claiming of storage devices -- particularly, fiber channel storage
10 devices -- other than those so identified.

Further aspects of the invention provide a SAN as described above which provides for blocking access to, or masking, a storage device to which access had previously not been blocked. According to these aspects, the agent or other functionality (e.g., resident on the host) masks the storage device by invalidating a disk object previously created for that device. The device can later be unmasked, e.g., in response to an operator/administrator request, by validating that disk
15 object.

Still further aspects of the invention provide a SAN as described above which provides for
20 unmasking a storage device to which access had previously been masked. According to these aspects of the invention, the filter responds to the manager's identifying such a storage device to be unmasked by invoking the port driver for purposes of claiming the one or more storage devices identified by it as coupled to the selected digital data processor. In this regard, the filter